Supplement for Differential Peak Calling of ChIP-seq Signals with Replicates with THOR

Manuel Allhoff, Kristin Seré, Juliana F. Pires, Martin Zenke and Ivan G. Costa

1 Hidden Markov Model Estimates

We follow the definition of Couvreur [1] to formalize the Baum-Welch algorithm. We have to solve following optimization problem

$$\hat{B} \in \arg\max_{B \in \overline{B}} \sum_{s=0}^{S} \sum_{j=0}^{L} r_{sj} \log b_s(x_{\cdot j}), \tag{1}$$

where S gives the HMM's states, L the dimension of the data $\mathbf{X}$, r_{sj} the posterior probability, b_s the probability mass function for state s, $x_{\cdot j}$ the genomic signal for bin j and $\overline{B}$ the set from where we have to choose the free parameters. In our case, the probability mass function b_s is given by

$$b_s(x_{\cdot j}) = \prod_{k \leq |G|} \prod_{i \in G_k} g(x_{ij} | \Theta_{sG_k}),$$

where G contains the sets of all experiments, G_k contains all indices that belong to condition k, and Θ_{sG_k} are the unknown parameters associated to function g. We choose a Negative Binomial distribution (Ismail and Jemain [2]) as emission distribution g, that is,

$$g(x_{ij} | \Theta_{sG_k}) = \frac{\Gamma(x_{ij} + a_{sG_k}^{-1})}{\Gamma(x_{ij}+1) \cdot \Gamma(a_{sG_k}^{-1})} \cdot \left(\frac{a_{sG_k}^{-1}}{a_{sG_k}^{-1} + \mu_{sG_k}} \right)^{a_{sG_k}^{-1}} \cdot \left(\frac{\mu_{sG_k}}{a_{sG_k}^{-1} + \mu_{sG_k}} \right)^{a_{sG_k}^{-1}}, \tag{2}$$

with free parameters $\Theta_{sG_k} = \{a_{sG_k}, \mu_{sG_k}\}$, where a_{sG_k} is the dispersion parameter and where μ_{sG_k} gives the location.

In our case, we have $|G| = 2$ conditions and $S = 3$ HMM's states. Given Equation 1, we therefore have to solve 6 optimization problems to determine $\hat{B}$, that is, determining Θ_{sG_k} for each condition and state. As described in the main document, we follow a moment approach and restrict our estimates to μ_{sG_k}. We first compute μ_{sG_k}, and then use the mean variance function to estimate a_{sG_k} from μ_{sG_k}. That is, we constrain our optimization space to $\Theta_{sG_k} = \{\mu_{sG_k}\}$. To avoid label switching problems in the HMM (Rabiner [3]), we furthermore restrict our HMM's emission, such that,

- $\mu_{1G_1} = \mu_{2G_2} = \mu_{\text{high}}$,
- $\mu_{1G_2} = \mu_{2G_1} = \mu_{\text{low}}$, and
- $\mu_{3G_1} = \mu_{3G_2} = \mu_{\text{low}}$.

Consequently, we only have to solve 2 optimization problems, that is, determining μ_{high} and μ_{low}, to solve Equation 1. Here we show the estimation of $\mu_{\text{high}} = \mu_{11} = \mu_{22}$. The other parameter estimates follow respectively.

We restrict our optimization space, such that μ_{high} only depends on $s = 1$ and $s = 2$. We then rewrite Equation 1 as

$$\begin{aligned}
&\arg\max_{\mu_{\text{high}}\in\Theta} \sum_{k\leq|G|}\sum_{i\in G_k}\sum_{j=0}^{L} r_{2j}\log g(x_{ij}|\mu_{2G_k}) + \sum_{k\leq|G|}\sum_{i\in G_k}\sum_{j=0}^{L} r_{1j}\log g(x_{ij}|\mu_{1G_k}) \\
=&\arg\max_{\mu_{\text{high}}\in\Theta} \sum_{i\in G_1}\sum_{j=0}^{L} r_{2j}\log g(x_{ij}|\mu_{2G_1}) + \sum_{i\in G_2}\sum_{j=0}^{L} r_{2j}\log g(x_{ij}|\mu_{2G_2}) \\
&+ \sum_{i\in G_1}\sum_{j=0}^{L} r_{1j}\log g(x_{ij}|\mu_{1G_1}) + \sum_{i\in G_2}\sum_{j=0}^{L} r_{1j}\log g(x_{ij}|\mu_{1G_2}) \\
=&\arg\max_{\mu_{\text{high}}\in\Theta} f(\mu_{\text{high}})
\end{aligned}$$

We define a function f depending on μ_{high}. As we want to optimize f, we derive f and obtain

$$\begin{aligned}
\frac{f}{\delta\mu_{\text{high}}} &= \frac{\sum_{i\in G_2}\sum_{j=0}^{L} r_{2j}\log g(x_{ij}|\mu_{2G_2})}{\delta\mu_{\text{high}}} &&+ \frac{\sum_{i\in G_1}\sum_{j=0}^{L} r_{1j}\log g(x_{ij}|\mu_{1G_1})}{\delta\mu_{\text{high}}} \\
&= \frac{f_1}{\delta\mu_{\text{high}}} &&+ \frac{f_2}{\delta\mu_{\text{high}}}
\end{aligned} \tag{3}$$

Sums containing μ_{2G_1} and μ_{1G_2} are constants while deriving f with regard to μ_{high} and therefore are no longer considered. To simplify the notation, we introduce functions f_1 and f_2, which we have to derive separately to obtain the derivation of f.
The derivation estimation for f_2 works respectively. Accordingly to Ismail et al. [2], we can rewrite Equation 2 as

$$\begin{aligned}
g(x_{ij}|\Theta_{sG_k}) = &\left(\sum_{h=1}^{x_{ij}-1}\ln(1+a_{sG_k}h)\right) - x_{ij}\cdot\ln(a_{sG_k}) - \ln(x_{ij}!) + x_{ij}\cdot\ln(a_{sG_k}\cdot\mu_{sG_k}) \\
&- (x_{ij}+a_{sG_k}^{-1})\cdot\ln(1+a_{sG_k}\cdot\mu_{sG_k})
\end{aligned} \tag{4}$$

We plug in Equation 4 in function f_1 of Equation 3. The derivation of f_1 is given by

$$\frac{f_1}{\delta\mu_{\text{high}}} = \sum_{i\in G_2}\sum_{j=0}^{L} r_{ij}\frac{x_{ij}-\mu_{\text{high}}}{\mu_{\text{high}}+a_1\mu_{\text{high}}^2}$$

We plug in $f_1/\delta\mu_{\text{high}}$ and $f_2/\delta\mu_{\text{high}}$ in Equation 3, set $f/\delta\mu_{\text{high}}$ to 0 and obtain the parameter $\hat{\mu}_1$ that optimize function f. That is, we write

$$\begin{aligned}
\frac{f}{\delta\mu_{\text{high}}} \stackrel{!}{=} 0 &= \sum_{i\in G_2}\sum_{j=0}^{L} r_{ij}\frac{x_{ij}-\mu_{\text{high}}}{\mu_{\text{high}}+a_1\mu_{\text{high}}^2} + \sum_{i\in G_1}\sum_{j=0}^{L} r_{ij}\frac{x_{ij}-\mu_{\text{high}}}{\mu_{\text{high}}+a_1\mu_{\text{high}}^2} \\
\Rightarrow \qquad \hat{\mu}_1 &= \frac{\sum_{i\in G_2}\sum_{j=0}^{L} r_{2j}x_{ij} + \sum_{i\in G_1}\sum_{j=0}^{L} r_{1j}x_{ij}}{|G_2|\sum_{j=0}^{L} r_{2j} + |G_1|\sum_{j=0}^{L} r_{1j}}
\end{aligned}$$

Parameter $\hat{\mu}_{\text{low}}$ is computed accordingly.

2 ChIP-seq Simulator with Replicates

Evaluation and comparison of differential peak callers is still an open problem. There are no datasets which can serve as a gold standard in the evaluation procedure. To overcome the lack of ground truth simulated datasets with true positive peaks can be designed. The simulation of single ChIP-seq datasets has already been addressed [4, 5, 6]. However, these approaches either cannot be directly used in the differential peak calling problem as they focus on single ChIP-seq signals [4, 5], or are not freely available and does not parametrize the variance between replicates [6]. Therefore, we developed an algorithm inspired by Humburg [5] to generate ChIP-seq reads simulating a pair of biological conditions with differential peaks [7]. Here we extensively extend improved our previous approach [7] to deal with replicates.

2.1 Method

For a given reference genome the procedure is: (1) selecting genomic regions to include protein domains (set of neighboring binding proteins), and sampling the number of proteins in a domain, (2) sampling and placement of fragments per protein, (3) assigning fragments to a replicate and a biological condition, (4) adding noise to the data and (5) deriving reads from the fragments and defining differential peaks (DPs). We use the original position of the proteins and the proportion of reads to define DPs. Figure 3 pictures the workflow of the simulation.

Compared to our previous approach [7], we comprehensively expanded our simulation algorithm. In regard to step (1), we refine the estimation of the space between proteins within the same domain by using empirical data on histone positioning. Concering step (2), we improve the function determining the number of fragments per protein to obtain MA plot distributions resembling real ChIP-seq data. We also use now a Beta distribution for the allocation of reads to distinct replicates in step (3). Finally, step (4) is novel.

2.1.1 Creating Protein Domains

We define n protein domains $(D_i)_{i=1\ldots n}$ for a chromosome C. Repeated regions as well as unassembled parts of the genome are ignored. For each protein domain D_i, we sample the actual number q_i of proteins $(P_{i,j})_{j=1\ldots q_i}$ that are contained. The protein number q_i follows a Negative Binomial distribution $q_i \sim NB_{m_1,p_1}$. We determine the positions $r_{i,1}$ of the first protein $P_{i,1}$ by uniformly selecting a position within the chromosome: $r_{i,1} \sim U[C]$. We then place further proteins $r_{i,j}$ with a particular space between each other, that is, $r_{i,j} = r_{i,1} + \sum_{k=1}^{j-1} b_k$ $(j \in \{2 \ldots q_i\})$. The spacing variable b_k follows a mixture of normal distributions $b_k \sim \sum_l c_l \cdot N_{\mu_l,\sigma_l^2}$. Here, we extend our previous approach [7], where we only define a constant spacing between proteins.

2.1.2 Sampling Fragments

We sample the fragments $\{F_{i,j,l}\}$ that are bound to the protein P_{ij}. The length $s_{i,j,l}$ of each fragment $F_{i,j,l}$ follows a normal distribution $s_{i,j,l} \sim N_{\mu,\sigma^2}$. Fragments are assigned randomly to each DNA strand and always cover the entire length $o_{i,j}$ of the protein $P_{i,j}$ to which they are assigned to. However, since fragments are usually larger than the corresponding proteins, the fragments' midpoint $m_{i,j,l}$ is randomly moved up- or downstream. That is, $m_{i,j,l} = r_{i,j} + t$ with $t \sim U[-(s_{i,j,l} - o_{i,j}), (s_{i,j,l} - o_{i,j})]$.

The number l of fragments to sample is given by $l = f \cdot p$ where p follows a Negative Binomial distribution $p \sim NB_{m_2,p_2}$. MA plots of the distribution of read counts have a typical shape, that is, a non-linear decrease of the A values with an increase of M values. We model the non linearity by using factor f which is described by a Laplace function:

$$f_{b,\mu}(d_{i,j}) = \frac{1}{2b} \exp\left(-\frac{|d_{i,j} - \mu|}{b}\right), \tag{5}$$

with $b = 0.5$, $\mu = 0.2$ and where $d_{i,j}$ gives the ratio of fragments that are assigned to the first or second biological condition. Sup. Figure 4 shows an example of a MA plot of data samples with such parameters. The factor $f_{0.5,0.2}$ causes the typical non linear relationship between M and A values. Using factor f to compute the number l of fragments is a further difference to our simulator of ChIP-seq reads without replicates [7]. The simulator without replicates does not account for the non-linearity property of the MA-plot.

2.1.3 Assigning Fragments

For each $P_{i,j}$ the ratio $d_{i,j}$ follows a beta distribution $B(0.5, 0.5)$. Fragments of the first or second biological condition are then assigned to the replicates. The beta distribution $B(0.5, 0.5)$ is symmetrical to 0.5 and tends to assume the extreme values 0 and 1. We thereby increase the probability that fragments are mostly assigned to one signal which could potentially results in a DP. For each protein domain $P_{i,j}$ and each biological condition, we randomly choose a replicate and assign fragment $F_{i,j,l}$ to it.

For n replicates of one signal and for a constant vector $\overline{\alpha} = \langle \alpha_0, \ldots, \alpha_0 \rangle$ of length n where α_0 describes the variance to distribute fragments among the replicates, the probability distribution to assign fragments to replicates is given by a Dirichlet distribution of order n

$$f(\overline{x}, \overline{\alpha}) = \frac{1}{B(\alpha)} \prod_{i=1}^{K} x_i^{\alpha_i - 1},$$

with

$$B(\overline{\alpha}) = \frac{\prod_{i=1}^{K} \Gamma(\alpha_i)}{\Gamma\left(\sum_{i=1}^{K} \alpha_i\right)}.$$

For each fragment, we follow the sampled probabilities to assign it to a replicate. The lower α_0, the higher is the variance within the replicates. In our previous simulation approach, we only use a constant ratio to determine the assignment of fragments to ChIP-seq profiles. The use of beta distribution therefore is another extension of our current simulation algorithm.

2.1.4 Adding Noise

We follow Zhang et al. [4] to add noise to each replicate. We divide the genome into bins and assign a random weight to each bin. We assume that the majority of noise fragments in a ChIP-seq experiment appear in single locations, but some of them build dense clusters. We therefore use a right skewing gamma distribution to model a bins's weight.

Accordingly to the weights, we randomly sample t bins. Within each bin, we determine one fragment with a uniformly chosen position. The number t of chosen bins for replicate r is defined as

$$t = \min\left(\frac{\#\text{fragments}}{\text{FRiP}}, \frac{b \cdot \text{genome's length}}{\text{read's length}}\right).$$

FRiP is the fraction of reads in peaks. We use a FRiP of 5% which is the lowest threshold for ChIP-seq profiles recommended by Landt el al. [8]. To have the number t invariant towards genome's length, we multiply the ratio of genome's and read's length by b. The variable b gives the average background coverage.

2.1.5 Defining Differential Peaks

Reads are obtained by getting the initial u base pairs of fragments in the forward strand (or the last u base pairs of the reverse strand). We define a true DP for the first (second) signal when the number of fragment in the first (second) sample is higher than a given threshold e and at least v fragments are present in the first (second) signal, that is,

$$\frac{|\{F_{i,j,l}\}|_{\text{sample } i}|}{|\{F_{i,j,l}\}|} > e \quad \text{and} \quad \left|\{F_{i,j,l}\}\right|_{\text{sample } i} \geq v,$$

where $\{F_{i,j,l}\}|_{\text{sample } i}$ gives the fragments of sample i. The position of the DP is defined by the protein position $r_{i,j}$. We output the reads of each replicate in a fasta file.

2.2 Evaluation

2.2.1 Metric

A genomic region $r = (r_s, r_e)$ is described by its starting position r_s and ending position r_e. We omit the chromosome information as we restrict our analysis to one chromosome. The intersection of two genomic regions $r_1 = (r_{1s}, r_{1e})$ and $r_2 = (r_{2s}, r_{2e})$ is defined as

$$r1 \cap r2 = \begin{cases} (\max(r1_s, r2_s), \min(r1_e, r2_e)) & \text{if } r1 \text{ and } r2 \text{ overlap,} \\ \emptyset & \text{else.} \end{cases}$$

The subtraction of two genomic regions is defined as

$$r1 - r2 = \begin{cases} (r1_s, r2_s) & \text{if } r1 \text{ and } r2 \text{ overlap}, r1_s < r2_s, r1_e < r2_e, \\ (r2_e, r1_e) & \text{if } r1 \text{ and } r2 \text{ overlap}, r1_s > r2_s, r1_e > r2_e, \\ \{(r1_s, r2_s), (r2_e, r1_e)\} & \text{if } r1 \text{ and } r2 \text{ overlap}, r1_s < r2_s, r1_e > r2_e, \\ \emptyset & \text{if } r1 \text{ and } r2 \text{ overlap}, r1_s > r2_s, r1_e < r2_e, \\ (r1_s, r1_e) & \text{else.} \end{cases}$$

For two sets of genomic regions the subtraction and intersection operation is performed element-wise. The size of a genomic region set is defined as the sum of all genomic region's length.

Let T be the genomic region set of true positive DPs given by the simulation. Moreover, let $P_A = \{p_1, \ldots, p_m\}$ be the genomic region set of DPs that are predicted by algorithm A. Let

$$\hat{Y}_i = \frac{|p_i \cap T|}{|T|} \quad \text{and} \quad \hat{X}_i = \frac{|p_i - T|}{|\text{genome} - T|}$$

describe the ratio of the true and false called DPs respectively normalized against the size of true positive DPs and the genome. Element-wise addition of the p-value sorted list $D_i = \langle \hat{X}_i, \hat{Y}_i \rangle$ gives the j-th data point $\sum_{j \leq i} D_j$ of the plot.

2.2.2 Parameter Setting

One important parameter is the length between the proteins b_i. Since we are interested in modelling histones, we estimate mixture model parameters by using histone position data in yeast [9]. For this, we randomly take $10,000$ consecutive histone positions and fit a mixture normal distribution to their distance. We ignore positions which are 500bp away from each other, as we assume that these positions belong to two different histone domains. Bayesian information criterion (BIC) shows that 2 components fit best for the mixture model ($-1.5 \cdot 10^2$). We define the minimum distance between proteins/histones as the sum of the usual estimate of histone size (147bps) and the average linker size (55bps) (Szerlong and Hansen [10]).

We generate $n = 10,000$ protein domains per dataset. ChIP fragments typically have a length of 200 bp (Furey [11]). We therefore model the fragment's size with mean $\mu = 200$ and standard deviation $\sigma = 20$. The standard deviation follows estimates taken from paired-end sequencing data reported by Marschall et al. [12]. The minimum number of reads to support a DP v is 25 and the ratio e for definition of a DP is defined as $e = 0.6$. We use a typical read size u of 26. We use chromosome 1 of the mouse genome (mm9) as reference genome. Reads were aligned to chromosome 1 using BWA

with default parameters. In accordance with Landt et al. [8], the fraction of reads in peaks (FRiP) is 0.05. Our empirical studies have shown that the average background coverage b should be around 0.25 in ChIP-seq experiments. We use $m_1 = 8$ and $p_1 = 14$ for the Negative Binomial distribution NB_{m_1,p_1} describing the number of proteins in a protein domain. We repeat each experiment 25 times. Sup. Fig. 5 gives two examples for simulated ChIP-seq profiles.

3 Competing Methods

Here, we describe all differential peak callers which provide support for replicates. See Table 3 for an overview of the tool's characteristics. Some of these tools require results of a single signal peak caller. We use MACS2 on pooled replicates for this task.

Csaw Csaw (Aaron et al.[6]) main method is a based on a window-based approach to segment ChIP-seq profiles. A modified version of the TMM method is applied to normalize the CHIP-seq signal on 10kbp bins. EdgeR (Robinson et al.[13]), which is based on Negative Binomial distribution test, is used to assign a p-value to each differential peak. Latter, consecutive significant bins are merged to form final DPs followed by a correction of p-values following a sime's method. Input-DNA is not used to normalize ChIP-seq signals, but only in a postprocessing step to filter out potential DPs. Further, csaw does not normalize against `GC`-content and does not estimate the fragmentation size. As suggested by the authors, we use a window size of 150bp and a step size of 25bp. All other paprameters are set as default. We were not able to run CSAW on simulated data, even when trying out distinct parameters as used in the real data.

PePr PePr (Zhang et al. [14]) follows a window-based strategy to detect DPs. The windows size is automatically computed and equals the estimated average width of initially called peaks. PePr normalize the input-DNA to the mean of all ChIP-seq signals, computes the fold change of input-DNA and ChIP-seq signal and follows the TMM approach to globally normalize across different ChIP-seq profiles. PePr requires input-DNA to run. To check for DPs, first read counts are modelled by a Negative Binomial distribution and second Wald's test is applied to check for significance in read counts. Furthermore, PePr provides estimation of fragment size, input subtraction, filtering of peaks with strand bias, but does not correct for `GC`-content. We follow the instructions on their webpage (`https://ones.ccmb.med.umich.edu/wiki/PePr/`) including a procedure to remove artifacts in ChIP-seq data. To obtain a number of DPs comparable to other tools, we increase the p-value threshold parameter to 0.01.

MACS2 MACS2 (unpublished, available at `https://github.com/taoliu/MACS/`) works in two steps. First, all ChIP-seq profiles are pooled together and MACS2 SPC's algorithm (*callpeak*) is executed for each condition. Second, we use *bdgdiff* to identify DPs within these peaks. The SPC normalize against input-DNA and also considers `GC`-content. MACS2 differential peak method works by a sliding window approach on candidate regions (personal communication). There are no formal descriptions on its parameters and the strategy for normalization. Initially, MACS2 called too few DPs, such that we had to decrease both the minimum length for DPs by using $l = 50$ and the fold-change cutoff by using $C = 1.5$ in the algorithm bdgdiff. Moreover, we increased the p-value threshold to 0.2 to increase the number of peaks for the algorithm callpeak. For the simulated data, we used default parameters.

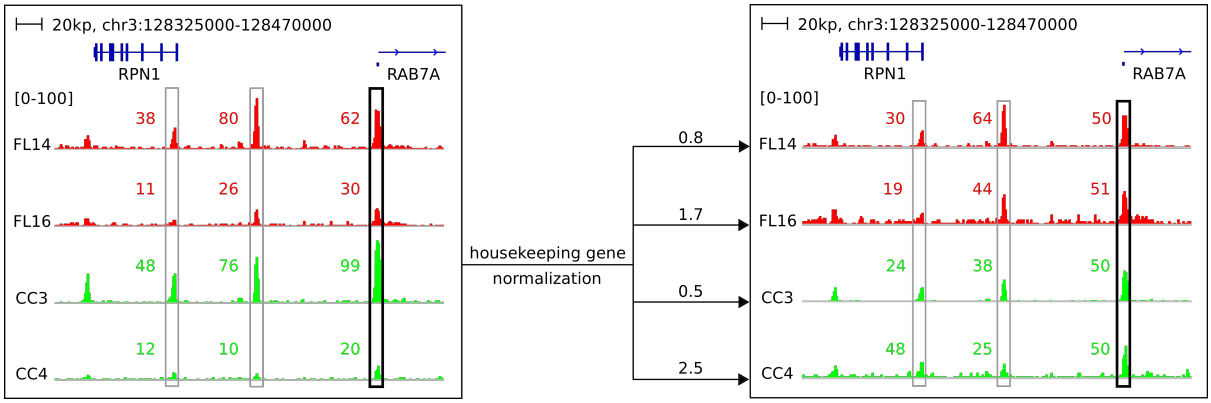
DiffBind DiffBind (Stark and Roy [15]) is a two-stage differential peak methods based on single peak candidate genes and edgeR. First, the peak lists are merged to obtain consensus peaks. The number of reads falling in to these consensus peaks are counted and a statistical model based on edgeR(Robinson et al.[13]) is estimated to call DPs. Normalization is done by TMM after input-control is subtracted from ChIP-seq profiles. Neither the fragmentation size nor `GC`-content is estimated by DiffBind. As recommended by authors, we use DiffBind with parameter *minOverlap* equals 3 in the count function to only consider peaks supported in up to three replicates across all conditions. Moreover, we increase the threshold for significant DPs (*th=0.1*).

DiffReps DiffReps (Shen et al. [16]) performs a sliding window approach to identify potential DPs. DiffReps globally normalizes by the geometric mean for each sample. Also, DiffReps takes input-DNA into account to normalize the ChIP-seq profiles. A pre-screening test ensures that only bins with a suffi-

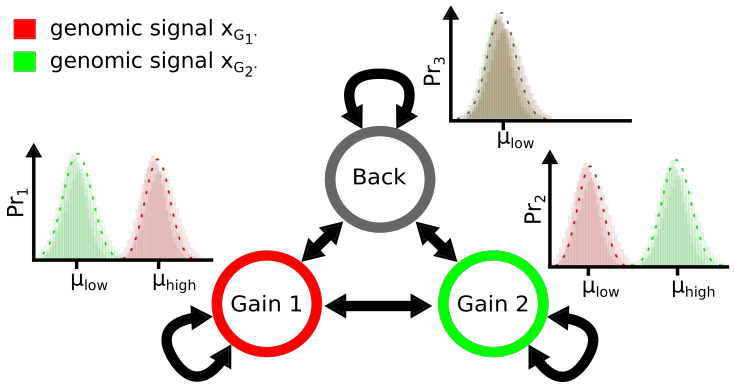
cient number of reads are taken into account. DiffReps uses a negative binomial test based on Anders et al. [17] to detect DPs. We run DiffReps with default parameters, that is, we use a window size if 1000bp and a step size of 100bp. We set the significance threshold for called DP (by using the option *–pval 0.1*).

DESeq-IDR Here we combine DESeq (Anders et al. [17]) and IDR(Landt el al. [8] and Li et al. [18]) to call DPs. DESeq is a tool to analysis differential gene expression and is commonly used to detect DPs (Liang et al. [19]). IDR is a method to define for a set of technical replicates a list of peaks with high consistency within the replicates. We follow the framework of ENCODE for the IDR computation (see `https://sites.google.com/site/anshulkundaje/projects/idr`). We use an IDR threshold of 0.01 for the replicates, an IDR threshold of 0.02 for the self-consistency replicates, and an IDR threshold of 0.0025 for the pooled pseudo replicates. We then apply DESeq with default parameters to check for DPs. DESeq takes the median of observed counts which are normalized with the geometric mean. Further, DESeq models the counts with a Negative Binomial distribution and uses these estimated functions to compute a p-value for each DP. We refer to this method as DESeq-IDR.

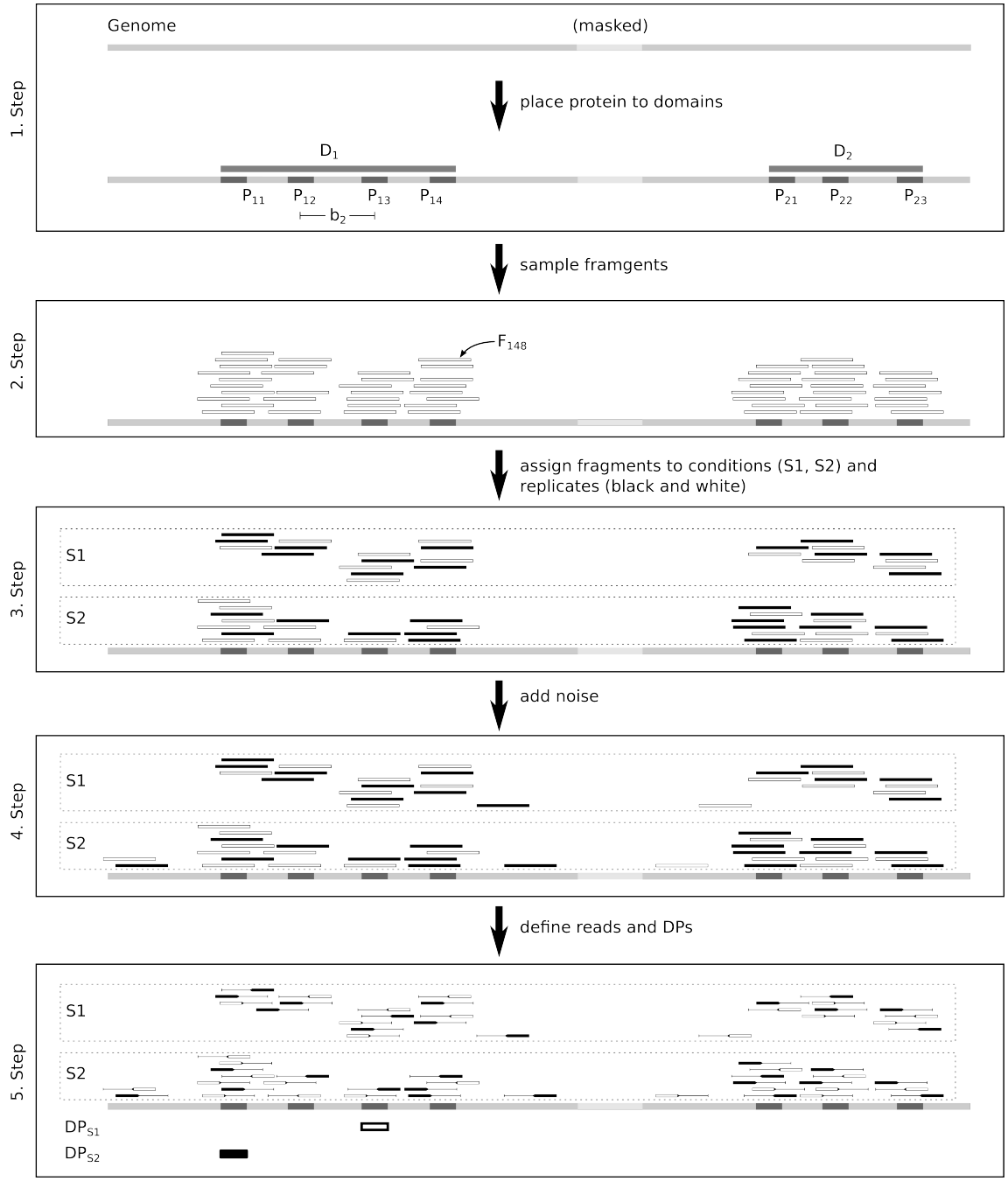
DESeq-JAMM We also use JAMM (Ibrahim et al. [20]), a recently published peak caller that takes replicates into account, to define a peak list for DESeq. We refer to this method as DESeq-JAMM. We use the SPC JAMM for our simulated datasets where we run it with default parameters. We were unable to execute JAMM on the biological data. JAMM takes input-DNA into account and subtracts it from ChIP-seq profiles. However, DESeq-JAMM does not apply any filter to avoid strand bias in DPs and does not take `GC`-content into account.



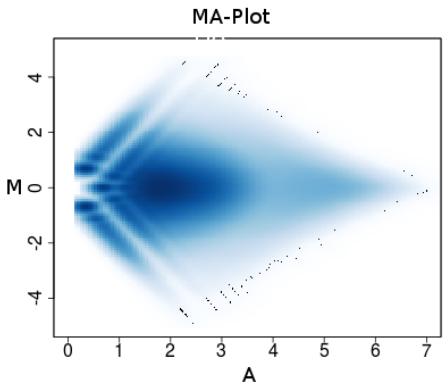
Supplementary Figure 1: Housekeeping genes normalization approach. The left panel shows two FL (red signal) and two CC (green signal) data set from LYMP. Boxes in signals contain peaks where its peak mass is given. The bold box is the promoter of a house keeping gene used used for the normalization. In this cartooned example, the normalization procedure gives 0.8 for FL14, 1.7 for FL16, 0.5 for CC3 and 2.5 for CC4 as normalization factor. The right panel shows the normalized signal with updated mass values of each peak located in a box. The housekeeping gene normalization approach brings all ChIP-seq signals to the same scale for any further downstream analysis steps.



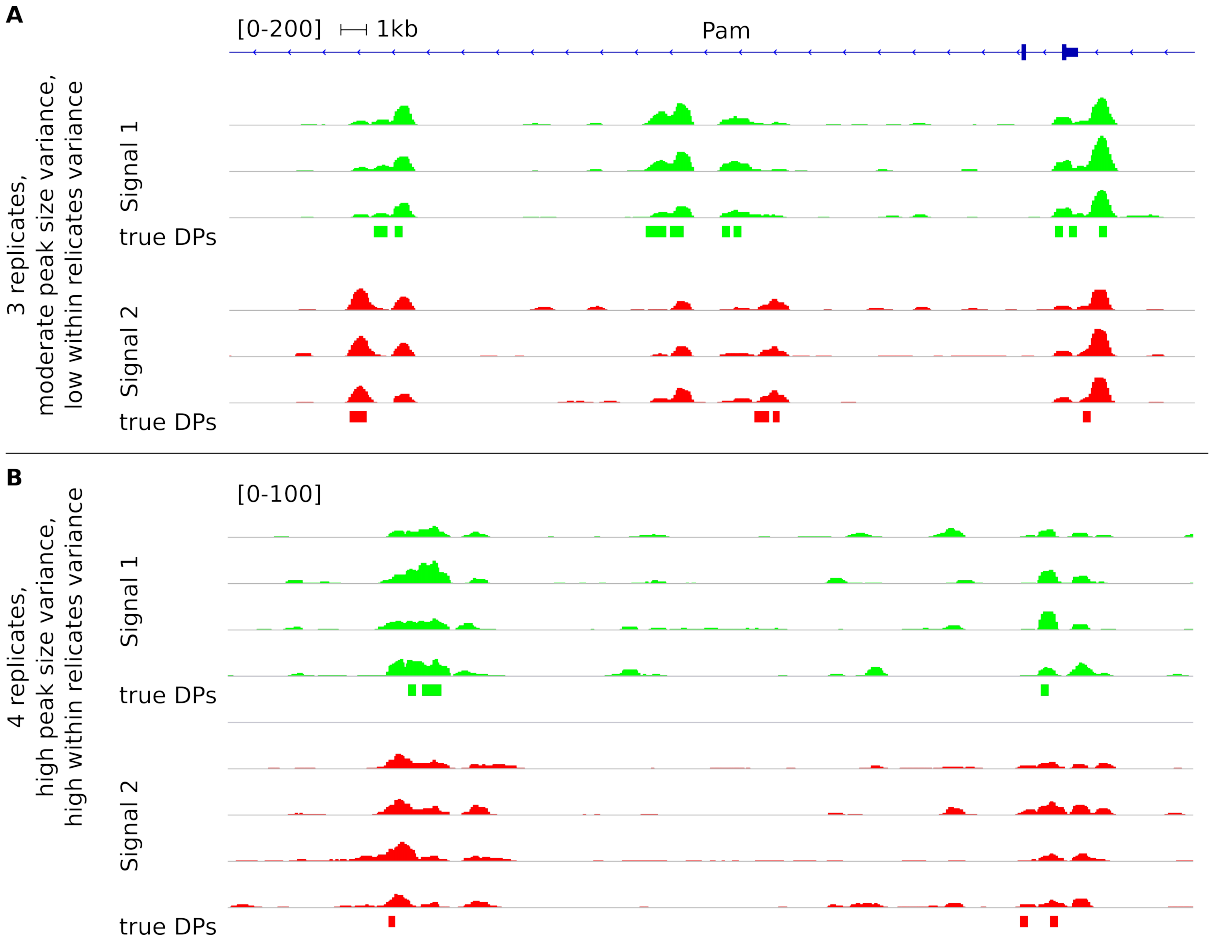
Supplementary Figure 2: Hidden Markov Model to identify DPs. The emission distributions (dotted lines) are assigned to each state and are based on Negative Binomial distributions. To avoid label switching and reduce number of free parameters, we constraint several parameters of the emission distributions. That is, the location parameter associated to gain peaks are equal $\mu_{1G_1} = \mu_{2G_2}$ (μ_{high}), as well as the location parameter associated to lost peaks and background states $\mu_{2G_1} = \mu_{1G_2}$ and $\mu_{3G_1} = \mu_{3G_2}$ (μ_{low}).



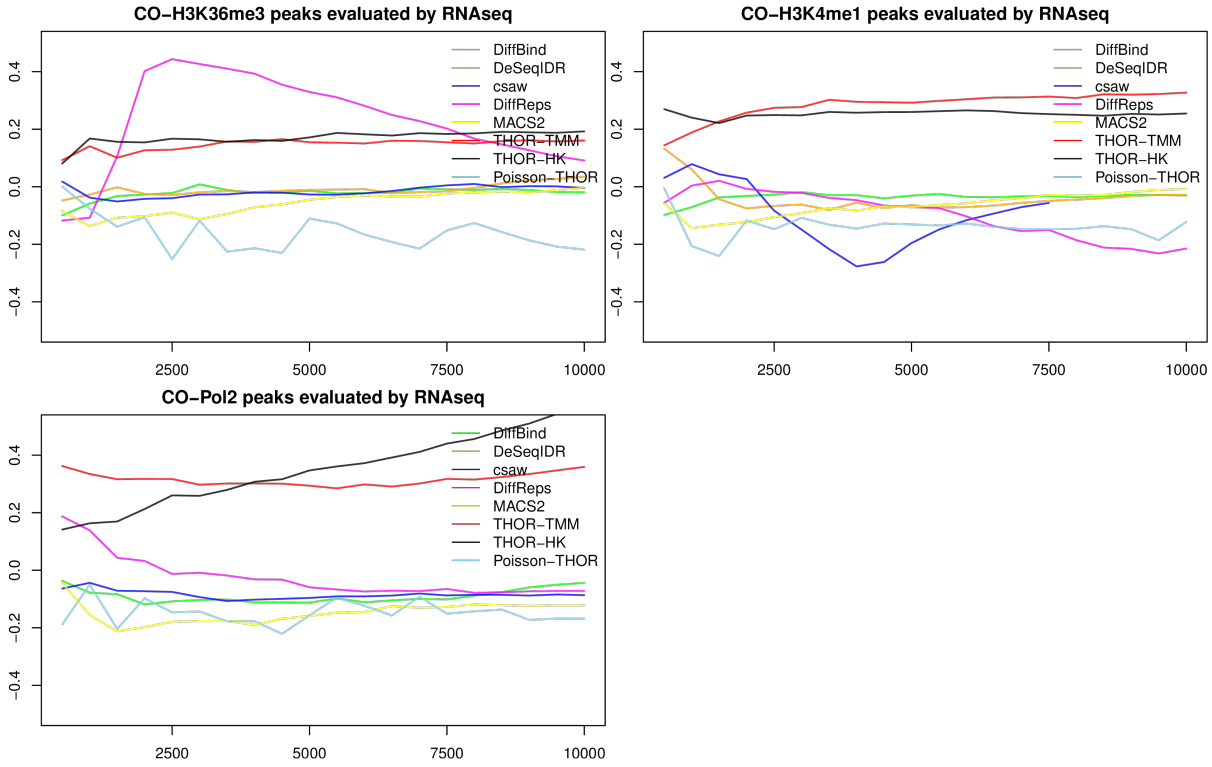
Supplementary Figure 3: Workflow to simulate ChIP-seq data. First, unassembled and repeated regions are marked and ignored in the further progress. We then uniformly place domains of proteins in the genome. Here, domain D_1 contains proteins P_{11}, P_{12}, P_{13} and P_{14}, and Domain D_2 contains proteins P_{21}, P_{22} and P_{23}. The spacing between two proteins of a domain, e.g. b_2 between protein P_{12} and P_{13}, is sampled from a mixture normal distribution. Next, fragments are assigned to a protein, e.g. fragment F_{148} is associated to protein P_{14}. In the next step, fragments are assigned to both biological conditions ($S1$, $S2$) as well as replicates (black, white). We add noise to the data and define reads as the beginning or ending part of the fragments. We find a DP gaining $S1$ and another DP gaining $S2$ in domain D_1.



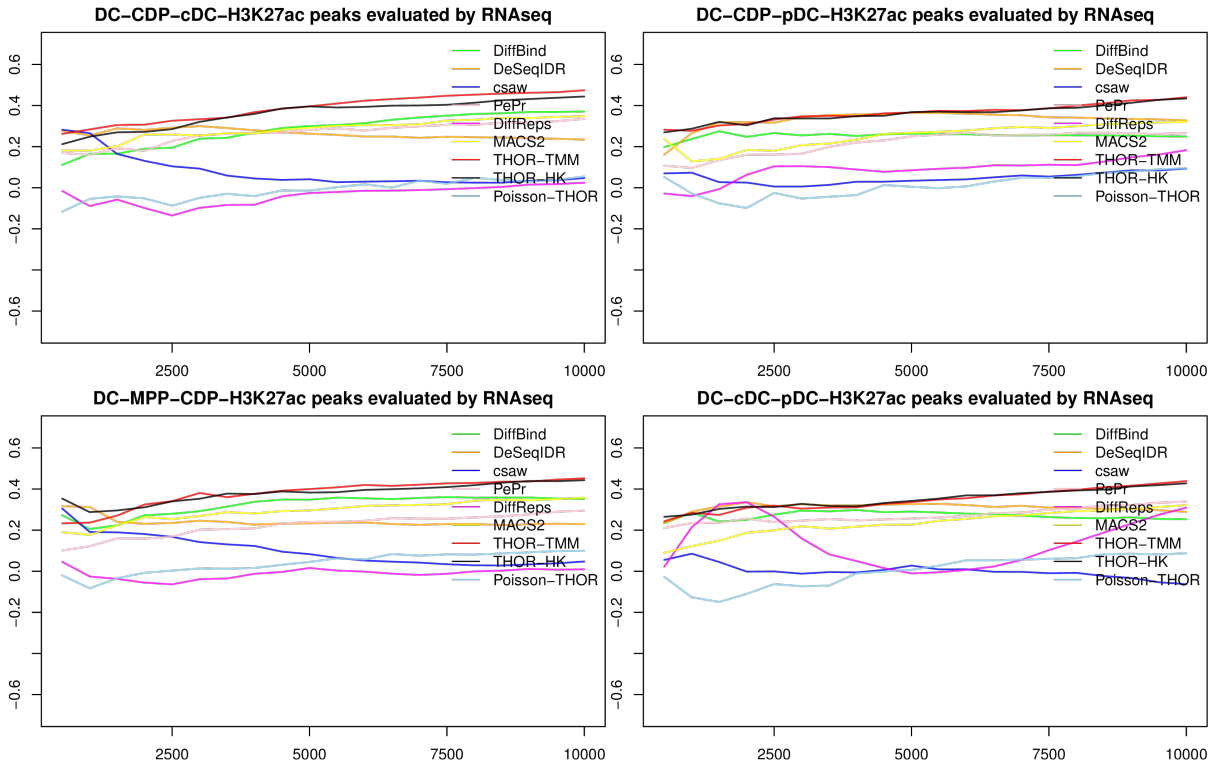
Supplementary Figure 4: MA-plot for simulated ChIP-seq data. We use mean $m_1 = 8$ and variance $p_1 = 14$ for the negative binomial distribution describing the protein domains. The number of fragments assigned to each protein follows a negative binomial distribution with mean $m_1 = 150$ and variance $p_1 = 10000$. We have 2 replicates for each condition with $\alpha_0 = 5$ for a moderate variance between the replicates. The use of the Laplace function (Equation 5) leads to the typical shape for the MA-plot.



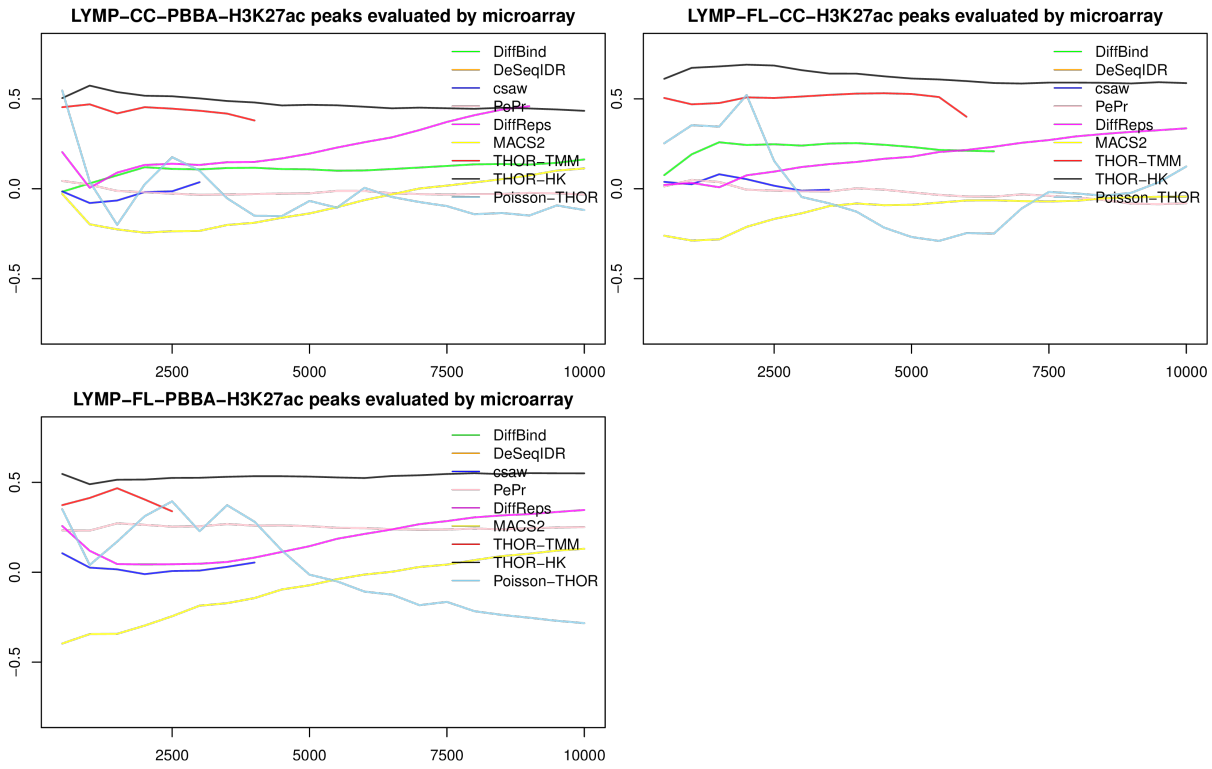
Supplementary Figure 5: Example for simulated data. **A)** An DPC problem with 3 replicates in each condition, moderate peak size variance and low within condition variance. **B)** A hard DPC problem with 4 replicates, high peak size variance and high within condition variance. We use the same among of reads for each simulated scenario, such that cases with low number of replicates contain more reads per replicates than cases with high number of replicates. Moreover, we show the true positive DPs for both scenarios.



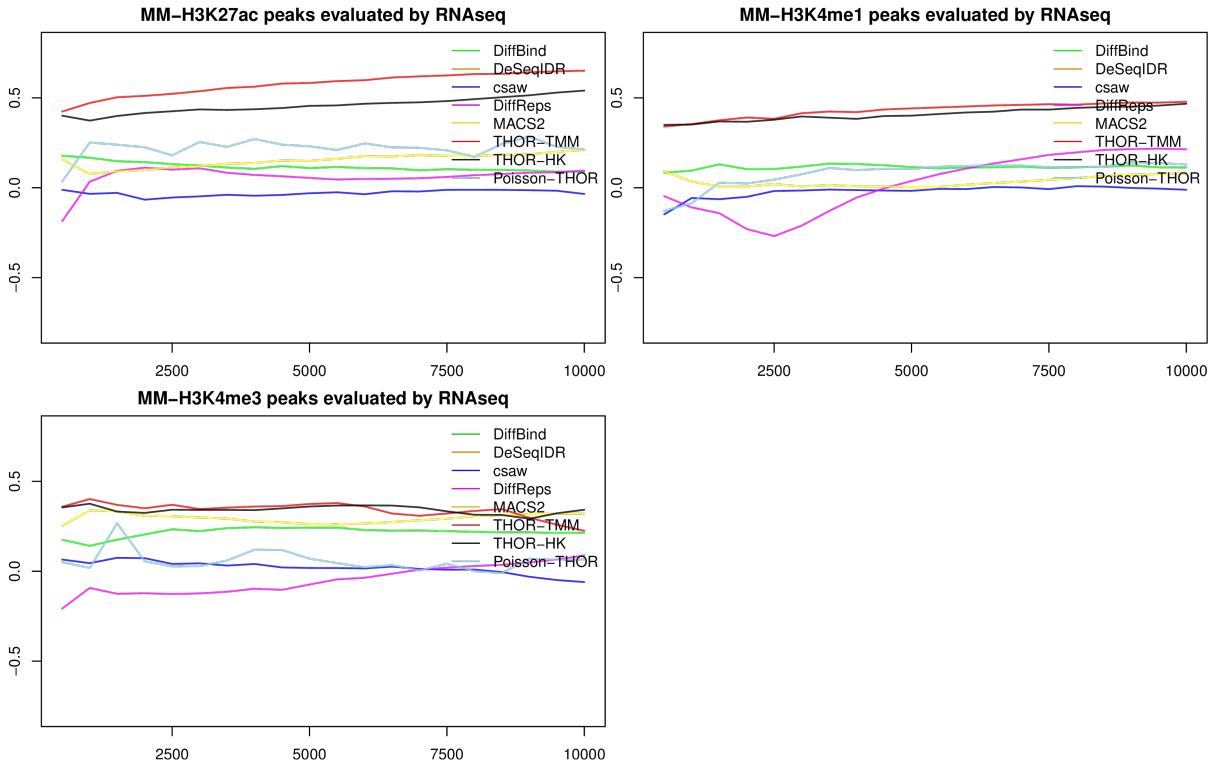
Supplementary Figure 6: Gene expression based DCA curves for CO study. PePr required input-DNA and is therefore unable to call DPs.



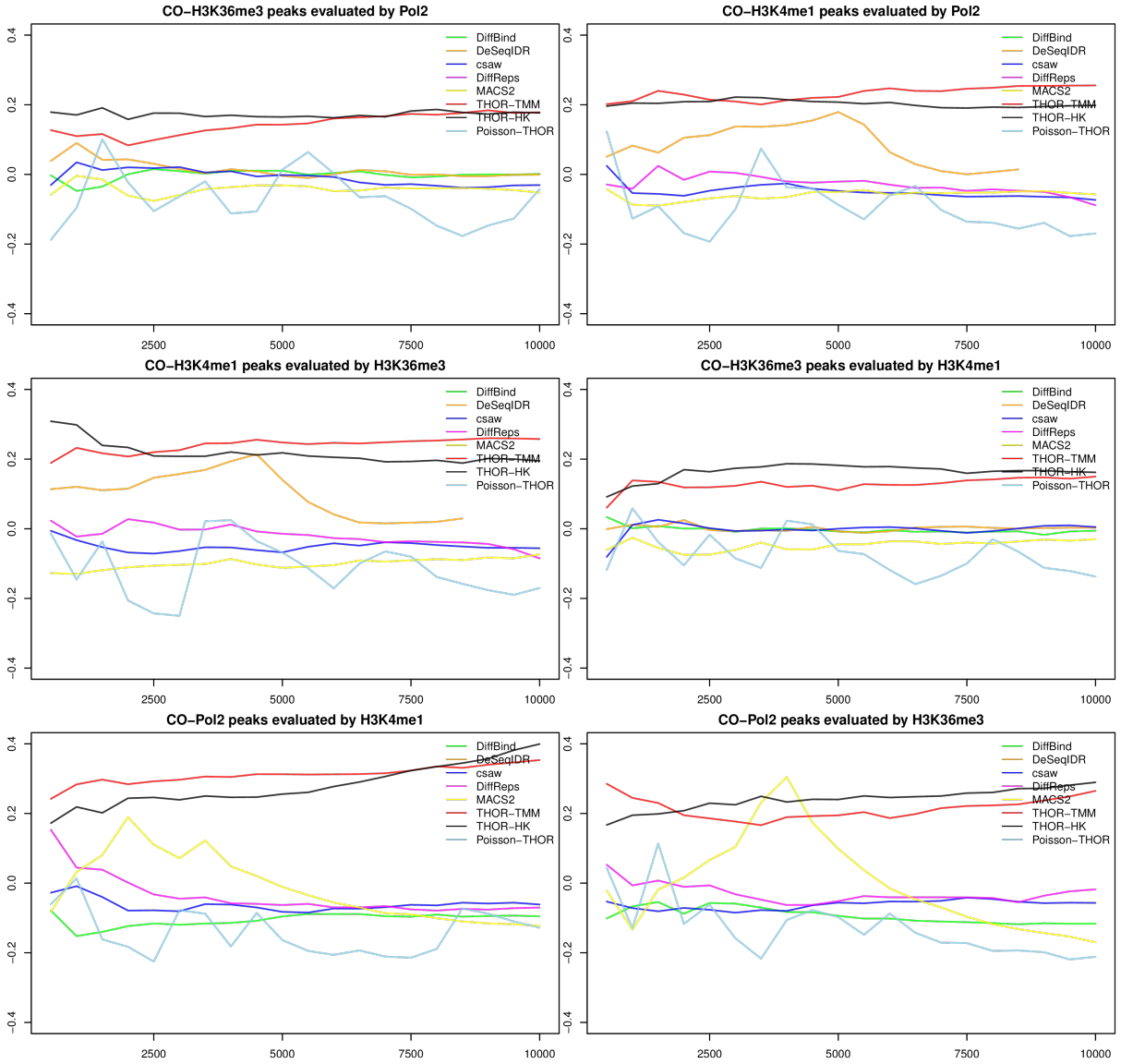
Supplementary Figure 7: Gene expression based DCA curves for DC study.



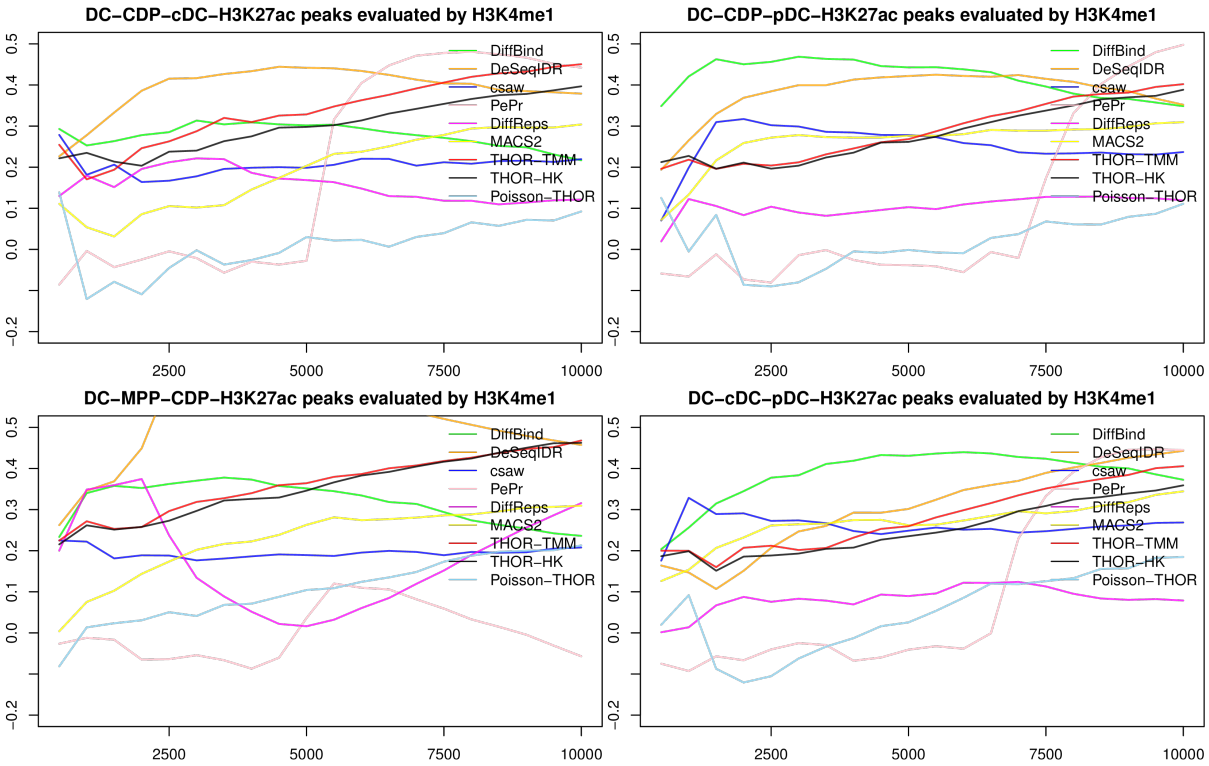
Supplementary Figure 8: Gene Expression based DCA curves for LYMP study. The DCA score is based on gene expression derived from microarray data.



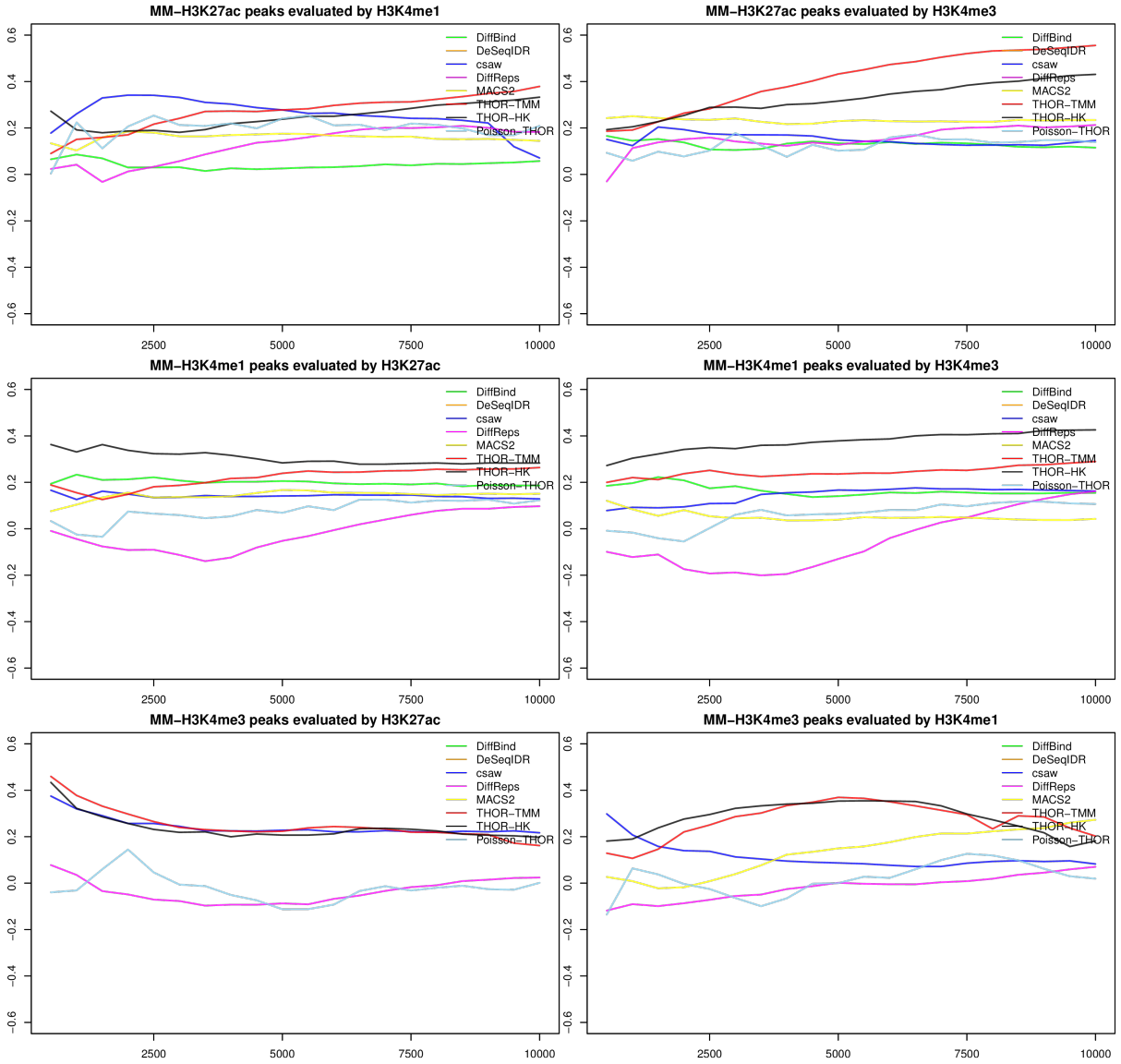
Supplementary Figure 9: Gene Expression based DCA curves for MM study. PePr required input-DNA and is therefore unable to call DPs.



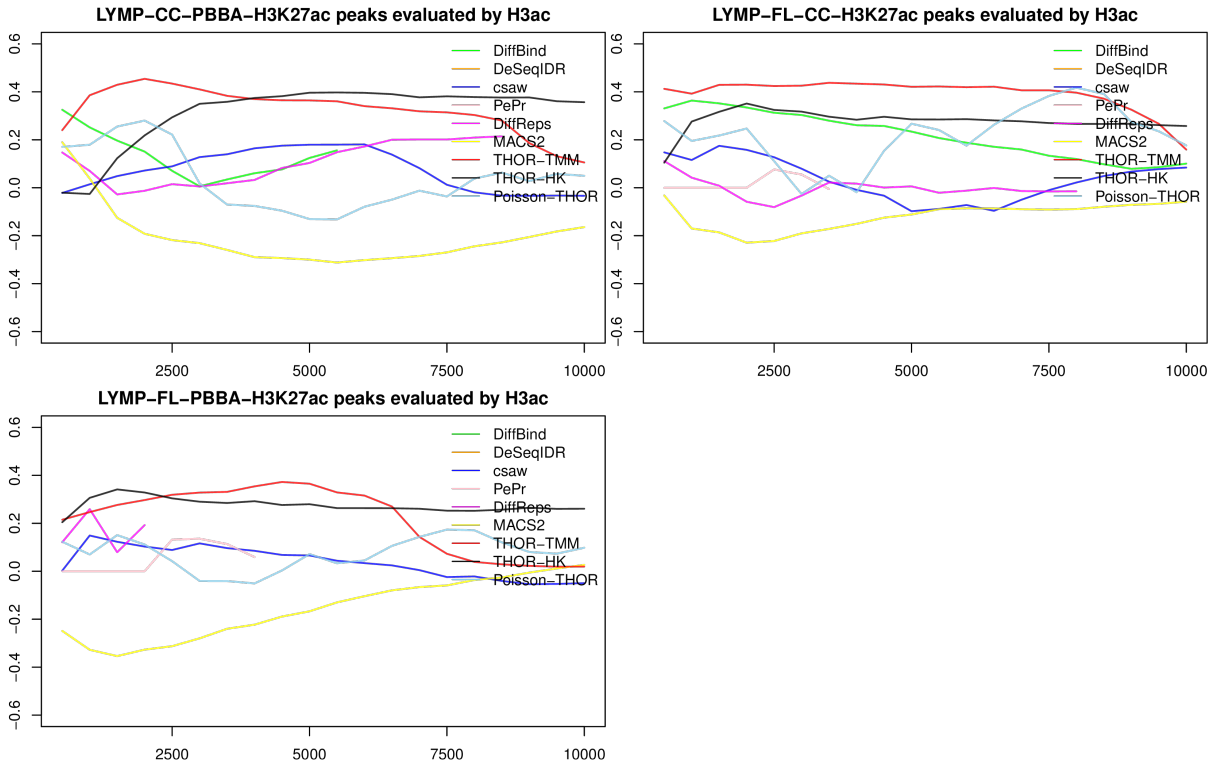
Supplementary Figure 10: Histone based DCA curves for CO study. The DCA score is based on a cross-validation with histones: H3K4me1 peaks evaluated with H3K36me3 and Pol2; H3K36me3 peaks evaluated with H3K4me1 and Pol2. PePr required input-DNA and is therefore unable to call DPs.



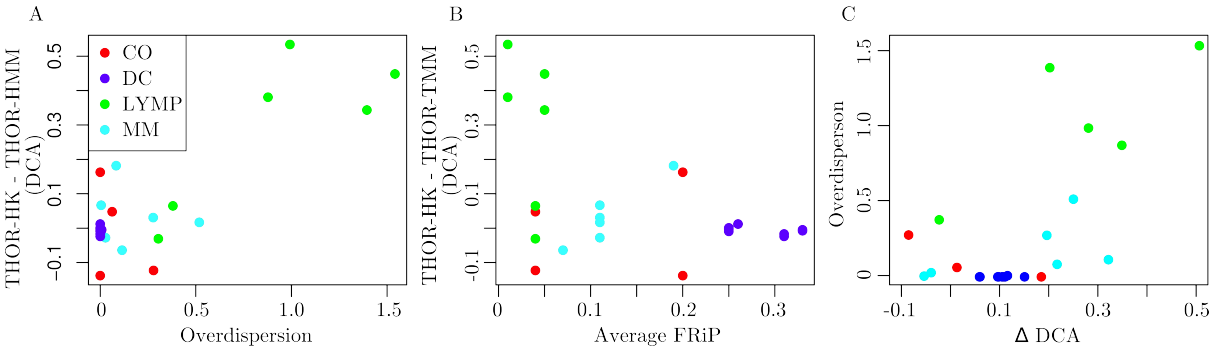
Supplementary Figure 11: Histone based DCA curves for DC study. Peaks were detected on H3K27ac and evaluated on H3K4me1 marks.



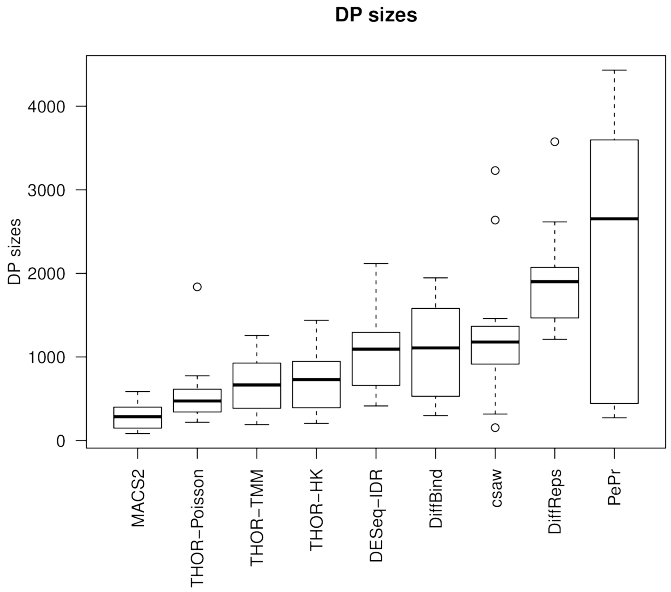
Supplementary Figure 12: Histone based DCA curves for MM study. The DCA score is based on a cross-validation with histones: H3K27ac evaluated with H3K4me1 and H3K4me3; H3K4me1 evaluated with H3K27ac and H3K4me3; and H3K4me3 evaluated with H3K27ac and H3K4me1.



Supplementary Figure 13: Histone based DCA curves for LYMP study. Peaks were detected on H3K27ac and evaluated with H3ac marks. PePr required input-DNA and is therefore unable to call DPs.



Supplementary Figure 14: Association between the difference in expression based DCA values for THOR normalization approaches (THOR-HK - THOR-TMM) vs. the overdispersion scores (A) and average FRIP (B). We also depict the association between the differences in expression based DCA scores of the best competing method and THOR and the overdispersion factor (C).



Supplementary Figure 15: Average DP sizes of each tool. The boxplot of each tool gives the DP size distribution obtained from predictions on all biological data.

AUC	
THOR-1.6/95	2.2857
THOR-1.3/95	2.4286
THOR-1.6/99	2.5
THOR-1.3/99	2.7857

Supplementary Table 1: Friedman ranking based on expression based DCA score ($h = 100, H = 1000$). We evaluate the initial parameter setting of THOR, that is, $t_1 \in \{\langle x \rangle^{.95}, \langle x \rangle^{.99}\}$ and $t_2 \in \{1.3, 1.6\}$ where t_1 is the fold change criteria and t_2 the minimum difference between signals based on percentile estimates (see main document Section 4.3.4 for details). The analysis is restricted to chromosome 1. For each metric, the methods are displayed in decreasing order with their respective Friedman ranking.

	THOR-1.6/95	THOR-1.3/95	THOR-1.6/99	THOR-1.3/99
THOR-1.6/95				
THOR-1.3/95				
THOR-1.6/99				
THOR-1.3/99				

Supplementary Table 2: Friedman-Nemenyi hypothesis test results for the expression based DCA score ($h = 100, H = 1000$) restricted to chromosome 1. The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

	One-Stage DPC	Segmentation Strategy	statistical model DP	frag. size estimation	input-DNA norm.	Subtracting input-DNA	GC-content	input-DNA not required	strand bias
THOR	×	HMM	NB	×	×	×	×	×	×
PePr	×	win	Wald's test	×	×	×			×
diffReps	×	win	NB		×			×	
csaw	×	win	NB					×	
MACS2		SPC	NA	×	NA	NA	NA	×	NA
DiffBind		SPC	NB			×		×	
DESeq-IDR		SPC	NB					×	
DESeq-JAMM		SPC	GMM, NB	×		×		×	

Supplementary Table 3: Tool's characteristics. Differential peak callers can be categorized in one-stage or two-stage approaches using either an HMM or a window-based approach to segment the ChIP-seq profiles. They perform a statistical test based on a Negative Binomial (NB) distribution, Wald's test or Gaussian mixture model (GMM) to identify DPs. Input-DNA can be normalized and may be used to subtract it from ChIP-seq profiles. Also, normalizing against GC-content may prohibit bias in profiles. For DESeq-JAMM, JAMM uses GMM to detect peaks and DESeq uses NB to detect DPs. JAMM subtracts the input-DNA from ChIP-seq profiles.

AUC	
THOR	1.0652
MACS2	3.0598
DESeq-JAMM	3.9185
DiffReps	4.087
DESeq-IDR	4.4891
DiffBind	5.0761
Poisson-THOR	6.3043

Supplementary Table 4: Friedman ranking of simulated data for all parameter settings based on the AUC statistic (see main document Section 4.3.3 for details). The methods are displayed in decreasing order with their respective Friedman ranking.

	THOR	MACS2	DESeq-JAMM	DiffReps	DESeq-IDR	DiffBind	Poisson-THOR
THOR							
MACS2	$*$						
DESeq-JAMM	$*$	$+$					
DiffReps	$*$	$*$					
DESeq-IDR	$*$	$*$					
DiffBind	$*$	$*$	$*$	$*$			
Poisson-THOR	$*$	$*$	$*$	$*$	$*$	$*$	

Supplementary Table 5: Friedman-Nemenyi test results based on the AUC statistic of simulated data for all scenarios. The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

	THOR	DESeq-IDR	MACS2	DiffReps	DiffBind	DESeq-JAMM	Poisson-THOR
THOR							
DESeq-IDR							
MACS2							
DiffReps	$*$						
DiffBind	$*$	$*$	$+$				
DESeq-JAMM	$*$	$*$	$*$				
Poisson-THOR	$*$	$*$	$*$	$*$			

Supplementary Table 6: Friedman-Nemenyi hypothesis test results for the **AUC** metric. We consider the case with 2 replicates, low within condition variance, and moderate peak size variability. The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

	THOR	DESeq-IDR	MACS2	DiffReps	DiffBind	DESeq-JAMM	Poisson-THOR
THOR							
DESeq-IDR							
MACS2							
DiffReps	$*$						
DiffBind	$*$	$*$	$+$				
DESeq-JAMM	$*$	$*$	$*$				
Poisson-THOR	$*$	$*$	$*$	$*$			

Supplementary Table 7: Friedman-Nemenyi hypothesis test results for the **AUC** metric. We consider the case with 2 replicates, medium within condition variance, and moderate peak size variability. The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

	THOR	DESeq-IDR	MACS2	DiffReps	DiffBind	DESeq-JAMM	Poisson-THOR
THOR							
DESeq-IDR							
MACS2							
DiffReps	$*$	$+$					
DiffBind	$*$	$*$					
DESeq-JAMM	$*$	$*$	$*$				
Poisson-THOR	$*$	$*$	$*$				

Supplementary Table 8: Friedman-Nemenyi hypothesis test results for the **AUC** metric. We consider the case with 2 replicates, high within condition variance, and moderate peak size variability. The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

	THOR	DESeq-IDR	MACS2	DiffReps	DESeq-JAMM	Poisson-THOR	DiffBind
THOR							
DESeq-IDR							
MACS2							
DiffReps	$*$	$+$					
DESeq-JAMM	$*$	$*$					
Poisson-THOR	$*$	$*$	$*$				
DiffBind	$*$	$*$	$*$	$*$			

Supplementary Table 9: Friedman-Nemenyi hypothesis test results for the **AUC** metric. We consider the case with 2 replicates, low within condition variance, and high peak size variability. The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

	THOR	DESeq-IDR	MACS2	DiffReps	DESeq-JAMM	Poisson-THOR	DiffBind
THOR							
DESeq-IDR							
MACS2							
DiffReps	$*$						
DESeq-JAMM	$*$	$*$	$+$				
Poisson-THOR	$*$	$*$	$*$				
DiffBind	$*$	$*$	$*$	$*$			

Supplementary Table 10: Friedman-Nemenyi hypothesis test results for the **AUC** metric. We consider the case with 2 replicates, medium within condition variance, and high peak size variability. The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

	THOR	DESeq-IDR	MACS2	DiffReps	Poisson-THOR	DESeq-JAMM	DiffBind
THOR							
DESeq-IDR							
MACS2	$*$						
DiffReps	$*$	$+$					
Poisson-THOR	$*$	$*$					
DESeq-JAMM	$*$	$*$					
DiffBind	$*$	$*$	$*$	$+$			

Supplementary Table 11: Friedman-Nemenyi hypothesis test results for the **AUC** metric. We consider the case with 2 replicates, high within condition variance, and high peak size variability. The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

	THOR	DESeq-JAMM	MACS2	DiffReps	DiffBind	Poisson-THOR	DESeq-IDR
THOR							
DESeq-JAMM							
MACS2							
DiffReps	*						
DiffBind	*	*					
Poisson-THOR	*	*	*				
DESeq-IDR	*	*	*	*			

Supplementary Table 12: Friedman-Nemenyi hypothesis test results for the **AUC** metric. We consider the case with 4 replicates, low within condition variance, and moderate peak size variability. The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

	THOR	DESeq-JAMM	MACS2	DiffReps	DiffBind	Poisson-THOR	DESeq-IDR
THOR							
DESeq-JAMM							
MACS2							
DiffReps	*						
DiffBind	*	*					
Poisson-THOR	*	*	*				
DESeq-IDR	*	*	*	*			

Supplementary Table 13: Friedman-Nemenyi hypothesis test results for the **AUC** metric. We consider the case with 4 replicates, medium within condition variance, and moderate peak size variability. The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

	THOR	DESeq-JAMM	MACS2	DiffReps	DiffBind	Poisson-THOR	DESeq-IDR
THOR							
DESeq-JAMM							
MACS2							
DiffReps	∗						
DiffBind	∗	∗					
Poisson-THOR	∗	∗	∗				
DESeq-IDR	∗	∗	∗	∗			

Supplementary Table 14: Friedman-Nemenyi hypothesis test results for the **AUC** metric. We consider the case with 4 replicates, high within condition variance, and moderate peak size variability. The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

	THOR	DESeq-JAMM	MACS2	DiffReps	DiffBind	Poisson-THOR	DESeq-IDR
THOR							
DESeq-JAMM							
MACS2							
DiffReps	∗						
DiffBind	∗	∗					
Poisson-THOR	∗	∗	∗				
DESeq-IDR	∗	∗	∗	∗			

Supplementary Table 15: Friedman-Nemenyi hypothesis test results for the **AUC** metric. We consider the case with 4 replicates, low within condition variance, and high peak size variability. The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

	THOR	DESeq-JAMM	MACS2	DiffReps	DiffBind	Poisson-THOR	DESeq-IDR
THOR							
DESeq-JAMM							
MACS2							
DiffReps	*						
DiffBind	*	*					
Poisson-THOR	*	*	*				
DESeq-IDR	*	*	*	*			

Supplementary Table 16: Friedman-Nemenyi hypothesis test results for the **AUC** metric. We consider the case with 4 replicates, medium within condition variance, and high peak size variability. The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

	THOR	DESeq-JAMM	MACS2	DiffReps	DiffBind	Poisson-THOR	DESeq-IDR
THOR							
DESeq-JAMM							
MACS2							
DiffReps	*						
DiffBind	*	*					
Poisson-THOR	*	*	*				
DESeq-IDR	*	*	*	*			

Supplementary Table 17: Friedman-Nemenyi hypothesis test results for the **AUC** metric. We consider the case with 4 replicates, high within condition variance, and high peak size variability. The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

AUC	
THOR-HK	2.0
THOR-TMM	2.2667
macs2	4.2
DiffReps	4.6
DiffBind	4.7
DESeqIDR	5.2333
Poisson-THOR	6.2
csaw	6.8

Supplementary Table 18: Friedman ranking based on expression based DCA score ($h = 500, H = 10000$) for all datasets (CO, DC, LYMP and MM). The methods are displayed in decreasing order with their respective Friedman ranking.

	THOR-HK	THOR-TMM	macs2	DiffReps	DiffBind	DESeqIDR	Poisson-THOR	csaw
THOR-HK								
THOR-TMM								
macs2								
DiffReps	+							
DiffBind	+							
DESeqIDR	*	*						
Poisson-THOR	*	*						
csaw	*	*	+					

Supplementary Table 19: Friedman-Nemenyi hypothesis test results for the expression based DCA score ($h = 500, H = 10000$). The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

AUC	
THOR-HK	2.3333
THOR-TMM	2.6667
PePr	4.8333
DiffReps	4.8889
macs2	5.0
DiffBind	5.3889
DESeqIDR	5.5556
Poisson-THOR	7.1111
csaw	7.2222

Supplementary Table 20: Friedman ranking based on expression based DCA score ($h = 500, H = 10000$) for datasets DC and LYMP. We restrict the analysis to DC and LYMP as PePr requires input-DNA which is not provided by CO and MM. The methods are displayed in decreasing order with their respective Friedman ranking.

	THOR-HK	THOR-TMM	PePr	DiffReps	macs2	DiffBind	DESeqIDR	Poisson-THOR	csaw
THOR-HK									
THOR-TMM									
PePr									
DiffReps									
macs2									
DiffBind									
DESeqIDR									
Poisson-THOR	$*$	$*$							
csaw	$*$	$*$							

Supplementary Table 21: Friedman-Nemenyi hypothesis test results for the expression based DCA score ($h = 500, H = 10000$). The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

AUC	
THOR-TMM	1.8095
THOR-HK	2.0
DiffBind	4.1905
DESeqIDR	5.2381
csaw	5.2857
macs2	5.4286
DiffReps	5.7619
Poisson-THOR	6.2857

Supplementary Table 22: Friedman ranking based on the histone based DCA score ($h = 500, H = 10000$) for all datasets (CO, DC, LYMP and MM). The methods are displayed in decreasing order with their respective Friedman ranking.

	THOR-TMM	THOR-HK	DiffBind	DESeqIDR	csaw	macs2	DiffReps	Poisson-THOR
THOR-TMM								
THOR-HK								
DiffBind	$*$	$+$						
DESeqIDR	$*$	$*$						
csaw	$*$	$*$						
macs2	$*$	$*$						
DiffReps	$*$	$*$						
Poisson-THOR	$*$	$*$						

Supplementary Table 23: Friedman-Nemenyi hypothesis test results for the histone based DCA score ($h = 500, H = 10000$). The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

AUC	
THOR-TMM	2.3333
THOR-HK	2.6667
DiffBind	4.1667
csaw	5.3333
DiffReps	5.5556
DESeqIDR	5.8333
macs2	6.1111
Poisson-THOR	6.3333
PePr	6.6667

Supplementary Table 24: Friedman ranking based on histone based DCA score ($h = 500, H = 10000$) for datasets DC and LYMP. We restrict the analysis to DC and LYMP as PePr requires input-DNA which is not provided by CO and MM. The methods are displayed in decreasing order with their respective Friedman ranking.

	THOR-TMM	THOR-HK	DiffBind	csaw	DiffReps	DESeqIDR	macs2	Poisson-THOR	PePr
THOR-TMM									
THOR-HK									
DiffBind									
csaw									
DiffReps									
DESeqIDR									
macs2	$+$								
Poisson-THOR	$+$								
PePr	$*$	$+$							

Supplementary Table 25: Friedman-Nemenyi hypothesis test results for the histone based DCA score ($h = 500, H = 10000$). The asterisk and the cross, respectively, mean that the method in the column outperformed the method in the row with significance levels of 0.05 and 0.1.

References

[1] Couvreur, C. Hidden Markov Models and Their Mixtures. Diploma thesis Université catholique de Louvain Faculté des sciences – Département de mathématiques Belgium (1996).

[2] Ismail, N. and Jemain, A. A. (2007) Handling Overdispersion with Negative Binomial and Generalized Poisson Regression Models. *Casualty Actuarial Society Forum,*.

[3] Rabiner, L. R. (1989) A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE,* **77**(2), 257–286.

[4] Zhang, Z. D., Rozowsky, J., Snyder, M., Chang, J., and Gerstein, M. (2008) Modeling ChIP Sequencing In Silico with Applications. *PLoS Comput Biol,* **4**(8), e1000158.

[5] Humburg, P. ChIPsim: Simulation of ChIP-seq experiments (2011) R package version 1.18.0.

[6] Lun, A. T. L. and Smyth, G. K. (2014) De novo detection of differentially bound regions for ChIP-seq data using peaks and windows: controlling error rates correctly. *Nucleic Acids Research,* **42**(11), e95.

[7] Allhoff, M., Seré, K., Chauvistré, H., Lin, Q., Zenke, M., and Costa, I. G. (2014) Detecting differential peaks in ChIP-seq signals with ODIN. *Bioinformatics,* **30**(24), 3467–3475.

[8] Landt, S. G. et al. (2012) ChIP-seq guidelines and practices of the ENCODE and modENCODE consortia. *Genome Research,* **22**(9), 1813–1831.

[9] Weiner, A., Hughes, A., Yassour, M., Rando, O. J., and Friedman, N. (2010) High-resolution nucleosome mapping reveals transcription-dependent promoter packaging.. *Genome research,* **20**(1), 90–100.

[10] Szerlong, H. J. and Hansen, J. C. (2011) Nucleosome distribution and linker DNA: connecting nuclear function to dynamic chromatin structure.

[11] Furey, T. S. (2012) ChIP-seq and beyond: new and improved methodologies to detect and characterize protein-DNA interactions. *Nat Rev Genet,* **13**(12), 840–852.

[12] Marschall, T., Costa, I. G., Canzar, S., Bauer, M., Klau, G. W., Schliep, A., and Schönhuth, A. (2012) CLEVER: clique-enumerating variant finder.. *Bioinformatics,* **28**(22), 2875–2882.

[13] Robinson, M. D., McCarthy, D. J., and Smyth, G. K. (2010) edgeR: a Bioconductor package for differential expression analysis of digital gene expression data.. *Bioinformatics,* **26**(1), 139–140.

[14] Zhang, Y., Lin, Y.-H., Johnson, T. D., Rozek, L. S., and Sartor, M. A. (2014) PePr: a peak-calling prioritization pipeline to identify consistent or differential peaks from replicated ChIP-Seq data. *Bioinformatics,* **30**(18), 2568–2575.

[15] Stark, R. (2012) Differential Oestrogen Receptor Binding is Associated with Clinical Outcome in Breast Cancer.. In Chor, B., (ed.), *RECOMB*, Springer Vol. 7262 of Lecture Notes in Computer Science, p. 286.

[16] Shen, L., Shao, N.-Y., Liu, X., Maze, I., Feng, J., and Nestler, E. J. (2013) diffReps: Detecting Differential Chromatin Modification Sites from ChIP-seq Data with Biological Replicates. *PLoS ONE,* **8**(6), e65598+.

[17] Anders, S. and Huber, W. (2010) Differential expression analysis for sequence count data. *Genome Biology,* **11**(10), R106+.

[18] Li, Q., Brown, J. B., Huang, H., and Bickel, P. J. (2011) Measuring reproducibility of high-throughput experiments. *The Annals of Applied Statistics,* **5**(3), 1752–1779.

[19] Liang, K. and Keleş, S. (2012) Detecting differential binding of transcription factors with ChIP-seq. *Bioinformatics,* **28**(1), 121–122.

[20] Ibrahim, M. M., Lacadie, S. A., and Ohler, U. (2015) JAMM: a peak finder for joint analysis of NGS replicates. *Bioinformatics,* **31**(1), 48–55.